



# Isenbaev Number

Link submit: <http://acm.timus.ru/problem.aspx?space=1&num=1837>

Solution:

C++	<a href="http://ideone.com/Mkxa21">http://ideone.com/Mkxa21</a>
Java	<a href="https://ideone.com/NFti5U">https://ideone.com/NFti5U</a>
Python	<a href="https://ideone.com/vTxcdA">https://ideone.com/vTxcdA</a>

Tóm tắt đề:

Ta có một danh sách các người là bạn của nhau. Mỗi người đều có một tên riêng. Ta đặt đánh số nhân vật có tên "Isenbaev" là số 0, sau đó những người bạn quen trực tiếp với Isenbaev sẽ được đánh số 1, sau đó tiếp tục ta đánh số 2 cho những người bạn của bạn Isenbaev, cứ thế mà đánh số. Hỏi với từng người thì số tương ứng của họ là bao nhiêu.

Input:

Dòng đầu là một số nguyên dương  $n$  là số lượng mối quan hệ.

$n$  dòng sau, dòng thứ  $i$  chứa 3 chuỗi  $s_1, s_2, s_3$  tương ứng thể hiện tên của 3 người bạn với nhau.

Output:

In ra  $n$  dòng, mỗi dòng in một chuỗi và một số nguyên dương lần lượt là tên của một người và số tương ứng. Ta cần in ra các chuỗi theo thứ tự tăng dần theo bảng chữ cái alphabet.

Ví dụ:

7 Isenbaev Oparin Toropov Ayzenshteyn Oparin Samsonov Ayzenshteyn Chevdar Samsonov Fominykh Isenbaev Oparin Dublennykh Fominykh Ivankov Burmistrov Dublennykh Kurpilyanskiy Cormen Leiserson Rivest	Ayzenshteyn 2 Burmistrov 3 Chevdar 3 Cormen undefined Dublennykh 2 Fominykh 1 Isenbaev 0 Ivankov 2 Kurpilyanskiy 3 Leiserson undefined Oparin 1 Rivest undefined Samsonov 2 Toropov 1
--	--

### Hướng dẫn giải:

Rõ ràng nếu như ta xem như mỗi người là một đỉnh trên một đồ thị, thì việc đánh số cho một người có số là 0, sau đó ta đi đánh số đến những người bạn của người đó, thì rõ ràng ta có thể thấy ý tưởng thuật toán BFS lộ rõ ra ngay từ suy nghĩ này. Như vậy, ý tưởng của ta là ta sẽ quy định những người này là một đỉnh của đồ thị, sau đó sử dụng BFS để loang ra đánh số đỉnh tương ứng. Để có thể ánh xạ được từ một chuỗi sang một con số để có thể đánh số được đỉnh của đồ thị, ta sử dụng một cây nhị phân tìm kiếm key-value với key là string (tên mỗi người) và value là một con số ánh xạ tương ứng. Khi ta nhập một chuỗi s, nếu người có tên là s chưa được đánh số đồ thị (chưa xuất hiện trước đây) thì ta tiến hành đánh số đỉnh cho người đó.

**Độ phức tạp:**  $O(N \log N + N + M)$ , với việc ánh xạ từng tên sang số thì ta có độ phức tạp là  $O(N \log N)$  cho việc ánh xạ toàn bộ. Ngoài ra, thuật toán duyệt BFS của ta có độ phức tạp là  $O(N+M)$  với N là số lượng đỉnh và M là số lượng cạnh.